Control with Vergence Eye Movement in Augmented Reality See-Through Vision

Zhimin Wang [*] Yuxin Zhao [†] Feng Lu [‡]

State Key Laboratory of VR Technology and Systems, School of Computer Science and Engineering, Beihang University

ABSTRACT

Augmented Reality (AR) see-through vision has become a recent research focus since it enables the user to see through a wall and see the occluded objects. Most existing works only used common modalities to control the display for see-through vision, *e.g.*, button clicking and speech control. However, we use visual system to observe see-through vision. Using an addition interaction channel will distract the user and degrade the user experience. In this paper, we propose a novel interaction method using vergence eye movement for controlling see-through vision in AR. Specifically, we first customize eye cameras and design gaze depth estimation method for Microsoft HoloLens 2. With our algorithm, fixation depth can be computed from the vergence, and used to manage the see-through vision. We also propose two control techniques of gaze vergence. The experimental results show that the gaze depth estimation method is efficient. The difference cannot be found between these two modalities in terms of completion time and the number of successes.

Index Terms: Augmented Reality—See-through Vision—Vergence Eye Movement—; Human Computer Interaction (HCI)

1 INTRODUCTION

Augmented reality (AR) seeks to immerse the user via overlaying virtual objects onto the real environment [7]. AR also enables the user to see the hidden areas. This technique is called AR see-through vision [2], which lets the user see through a wall and see the occluded objects. More recent efforts have been made to improve the visualization of see-through vision. For instance, Avery *et al.* designed *Edge Overlay* technique to provide depth cues for see-through vision [2]. However, prior research mainly focused on the overlay effect of hidden areas and occluding layers (walls). They only used common interaction modalities to control the display of see-through vision in AR, such as button clicking and speech commands. However, we use visual system to observe the see-through vision. Using an addition interaction channel will distract the user and degrade the user experience. To this end, we propose using the gaze vergence to control the see-through vision in AR. We utilize the single modality to complete watching and controlling, *i.e.*, the eyes.

However, many AR devices do not support eye tracking. The recent Microsoft HoloLens 2 only offers the gaze direction but does not provide the gaze vergence or access to eye images. In this paper, we build a gaze tracking module with two infrared cameras, and assemble it into Microsoft HoloLens 2. Then we propose a gaze depth estimation algorithm for our modified Hololens. With our algorithm, fixation depth can be used to control the see-through vision. Overall, our contributions can be summarized as follows:

[*]joint first author, e-mail: zm.wang@buaa.edu.cn
[†]joint first author, e-mail: zyuxin@buaa.edu.cn
[‡]corresponding author, e-mail: lufeng@buaa.edu.cn

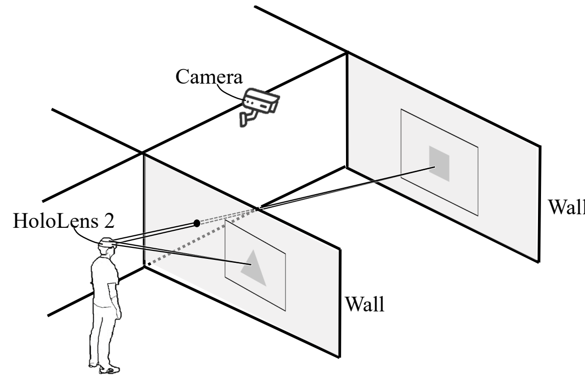

Figure 1: Control with vergence eye movement in AR see-through vision task. When the gaze vergence reaches the target depth behind the wall, the image captured by the surveillance camera is shown in HoloLens 2.

1. To the best of our knowledge, we are the first to use vergence eye movement to control see-through vision in AR.
2. We customize eye cameras and design gaze depth estimation method for HoloLens 2.
3. We propose two control techniques of gaze vergence for see-through vision, which are called stimulus-guided see-through technique and self-control see-through technique.

2 SYSTEM DESIGN

2.1 Gaze Depth Estimation

We control see-through vision via fixation depth. Prior research has studied computing the gaze depth, which can be classified into three different approaches. The first approach is learning the mapping from the x-coordinate offset of the gaze positions on the display plane [1, 5]. The another method is directly computing the intersection of the gaze rays from the two eyes [4]. The third approach use the Interpupillary Distance (IPD) to estimate the gaze depth. The first two methods need to gain the transformation between the customized eye trackers and AR display screen, which is difficult to get for HoloLens 2. We use the third method in that this way does not require the above transformation. The previous IPD-based methods mainly use the x-coordinate offset of pupil centers in pixels, which is directly measured from eye images. However, this method limits the depth estimation to 1.2 m due to the experimental error. We propose to compute the IPD in millimeters, which is based on 3D eyeball models. The distance range of depth estimation is improved to 4 m.

We first perform pupil tracking to compute the IPD. We employ the PuReST to detect pupil, which has the fast and robust performance [6]. We then perform the following computation: 1) Building the models of left and right eyes. Eye model fitting employs multi-frame pupil data in a continuous period of time. We use the latest

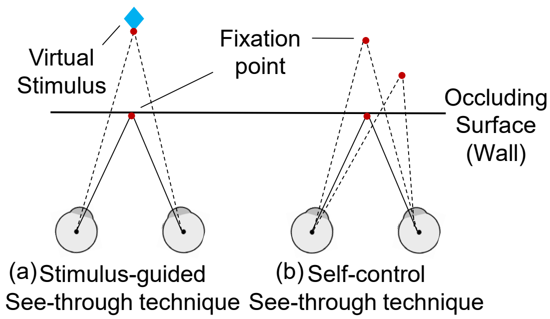

Figure 2: Two control techniques of gaze vergence when user looks towards the occluded surface. (a) Stimulus-guided See-through mode. (b) Self-control See-through mode.

proposed 3D eye model fitting method [3], which can mitigate the effects of corneal refraction and apply the two-sphere eye model. 2) Unifying the coordinate system of left and right eye cameras. The two cameras are placed into the same coordinate system by detecting a two-chessboard pattern in both cameras. Let p_l and p_r be the left and the right pupil centers. The IPD is computed in mm as $p_l - p_r$ in that coordinate system.

Our depth estimation range is from 0.5 m to 4 m. In order to reduce user calibration time, we only place three cubes in horizontal direction. These cubes will sequentially move at 0.33 m/s in AR HMD. We propose a piecewise depth fitting function to match the IPD data. Specifically, we first split these calibration data into three sets according to three calibration cubes. For each set, we employ the Random Sample Consensus (RANSAC) to extract inliers for polynomial regression. The degree of polynomial function is set to 2.

2.2 Two techniques of Controlling with Gaze Vergence

Most existing works only explored the vergence control technique under desktop or VR scenarios [1, 5]. We control the see-though vision via gaze vergence in AR. When user's point of regard (PoR) is located on the stimulus attached to the wall, we turn off the visualization of hidden scene. If the gaze vergence reaches the target depth behind the wall, the see-though vision can be activated. We are used to fixated on objects that are not occluded. However, we can converge lines of sight without the need for targets, *e.g.*, cross-eye technique [4]. In order to control the see-through vision comfortably, we propose two control modes of vergence eye movement:

1) Stimulus-guided see-through technique: a semi-transparent virtual stimulus is placed on user's gaze ray, which is located 5 meters away from the origin point, as shown in Fig. 2a. The user stands 1 meter from the wall. The user employs the stimulus as gaze guidance and thus the fixation depth increases for triggering the see-through vision. We empirically set the distance threshold and the time threshold as 1 meter and 1 second. The window of see-through vision always appears on the gaze ray and 2 meters behind the wall. This mode is similar to our viewing habit but the limitation is that this stimulus always appears on the gaze ray.

2) Self-control see-through technique: user freely controls the gaze vergence for seeing through walls without the need for targets, as shown in Fig. 2b. The user need to perform voluntary eye divergences by shrinking eye muscles. The distance threshold, the time threshold and the position of window are same as the first mode. This mode is natural and intuitive but need to practice for several times.

3 Results

We used the Microsoft HoloLens 2 as the AR HMD. The customized eye trackers were integrated with the HMD. Twelve participants, age 22 - 28 (9 males and 3 females) successfully completed the experiment. We evaluated the accuracy of gaze depth estimation. We used the distance between the estimated depth and the ground truth as the error metric. Quantitative results are shown in Table 1. Our method achieves 0.2 meter error (SD = 0.1) for near distance (<1 m). The error is within 1 meter for middle distance (<4 m). We think that this method is efficient for see-through vision control.

Table 1: The error of depth estimation (m). The first row represents the distance range. The second row includes the mean of the error and its standard deviation.

Distance	[0, 1]	[1, 2]	[2, 3]	[3, 4]
Error	0.2 ± 0.1	0.6 ± 0.3	0.9 ± 0.4	1.0 ± 0.3

Table 2: The comparison of two modalities on see-through vision task in terms of completion time and the number of successes.

Modalities / Metrics	Stimulus-guided Technique	Self-control Technique
Completion Time (s)	2.22 ± 0.72	2.14 ± 0.57
No. of Successes	9.92 ± 0.28	9.83 ± 0.55

We evaluated the efficiency of the two control techniques of gaze vergence in the see-through vision task. The users were asked to perform different actions according to the hints of "See Through Wall" and "See Wall" commands. These operations were done for ten times. We used the completion time and the number of successes as the evaluation metrics. The results are shown in Table 2. We employed the pairwise t-tests to judge whether the metrics are significantly different across modalities. We found there is no significant difference between these two techniques in terms of completion time and the number of successes.

4 Conclusion and Future Work

This research proposes a novel interaction using vergence eye movement to control see-through vision in AR. With our depth estimation algorithm, fixation depth can be computed from the vergence, and used to control the see-through vision. In the future, we plan to study the performance of different depths of the target. We will investigate the efficiency of different modalities for see-through vision control task, such as button clicking and speech control.

References

[1] S. Ahn, J. Son, S. Lee, and G. Lee. Verge-it: Gaze interaction for a binocular head-worn display using modulated disparity vergence eye movement. In *Conference on Human Factors in Computing Systems*, p. 1–7, 2020.

[2] B. Avery, C. Sandor, and B. H. Thomas. Improving spatial perception for augmented reality x-ray vision. In *IEEE Virtual Reality Conference*, pp. 79–82, 2009.

[3] K. Dierkes, M. Kassner, and A. Bulling. A fast approach to refraction-aware eye-model fitting and gaze prediction. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research Applications*, pp. 1–9, 2019.

[4] D. Kirst and A. Bulling. On the verge: Voluntary convergences for accurate and precise timing of gaze input. In *Conference on Human Factors in Computing Systems*, pp. 1519–1525, 2016.

[5] S. Kudo, H. Okabe, T. Hachisu, M. Sato, S. Fukushima, and H. Kajimoto. Input method using divergence eye movement. In *Conference on Human Factors in Computing Systems*, pp. 1335–1340, 2013.

[6] T. Santini, W. Fuhl, and E. Kasneci. Purest: Robust pupil tracking for real-time pervasive eye tracking. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research Applications*, ETRA '18, 2018.

[7] Z. Wang, H. Wang, H. Yu, and F. Lu. Interaction with gaze, gesture, and speech in a flexibly configurable augmented reality system. *IEEE Transactions on Human-Machine Systems*, 51(5):524–534, 2021.